

FRIDGE INSIGHT



JAVA PROJEKT :

HERR HAMZA HARTI.

HERR MAHMOUD MATAR.

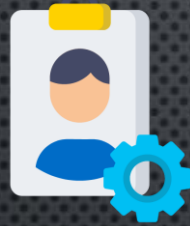
PROJEKT BESCHREIBUNG

- **FRIDGE-INSIGHT** IST EINE SOFTWARE, MIT DER SIE IHREN KÜHLSCHRANK BESSER VERWALTEN KÖNNEN. ES ENTHÄLT VIELE GUT PROGRAMMIERTE FUNKTIONEN, UM EIN OPTIMALES VERWALTUNGSSYSTEM ZU GEWÄHRLEISTEN.

MUSS-KRITERIEN

✓ BENUTZER VERWALTEN.

- KONTO ADDIEREN
- KONTO BEARBEITEN
- KONTO ENTFERNEN



✓ EINKAUF LISTE VERWALTEN

- EINKAUFSKORB ERSTELLEN
- ELEMENT ZUM EINKAUFSKORB HINZUFÜGEN
- ELEMENT VOM EINKAUFSKORB ENTFERNEN



✓ BENACHRICHTIGUNGEN KONFIGURIEREN

- BENACHRICHTIGUNGEN ERSTELLEN
- BENACHRICHTIGUNGEN BEARBEITEN
- BENACHRICHTIGUNGEN ENTFERNEN



✓ HINZUFÜGEN/ENTFERNEN EINES ELEMENTS

- PRODUKT ADDIEREN
- PRODUKT BEARBEITEN
- PRODUKT ENTFERNEN



WUNSCH-KRITERIEN

- ✓ DIE MENGE DER LEBENSMITTEL SOLLTE AUTOMATISCH AKTUALISIERT WERDEN.
- ✓ DIE BENACHRICHTIGUNGEN SOLLEN BEI LEBENSMITTELKNAPPHEIT AUTOMATISCH ERSTELLT WERDEN.
- ✓ JEDER BENUTZER HAT DIE MÖGLICHKEIT, EIN PROFIL ZU ERSTELLEN, IN DEM ER LIEBLINGSESSEN UND ANDERE FUNKTIONEN ABLEGEN KANN.

ABGRENZUNGS-KRITERIEN

- ✓ NUR DER "SUPERUSER" HAT DIE BERECHTIGUNG, EIN KONTO HINZUZUFÜGEN, ZU ÄNDERN ODER ZU ENTFERNEN.
- ✓ NUR EIN "SUPERUSER" KANN EIN PRODUKT HINZUFÜGEN, ÄNDERN ODER LÖSCHEN
- ✓ NUR EIN "SUPERUSER" KANN EIN ANDERES KONTO ALS "SUPERUSER" KENNZEICHNEN.

AKTEURE: BERECHTIGUNGEN UND BEGRENZUNGEN

Rolle	Beschreibung	Vertreter(Bsp.)	Bemerkungen
Super User	Der Superuser kann auf alle Funktionen des Programms zugreifen.	Vater/Mütter	
User	Der User kann auf alle grundlegenden Funktionen zugreifen	Kinder/Freunde	Vertreter kann Super User werden, aber er braucht die Erlaubnis eines der Super Users

FUNKTIONALE ANFORDERUNGEN



SuperUser

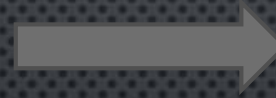


Die notigsten fünf Elemente

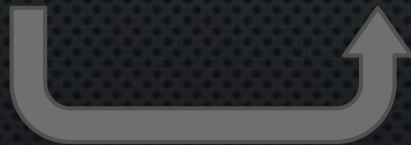


EINKAUFSKORB VERWALTUNG

PurchaseList		
f	listMaker	Account
f	dateOfPurchase	String
f	foodToBePurchased	LinkedList<Food>
m	PurchaseList()	
m	PurchaseList(Account, String, LinkedList<Food>)	
m	getFoodToBePurchased()	LinkedList<Food>
m	getListMaker()	Account
m	getDateOfPurchase()	String
m	setDateOfPurchase(String)	void
m	setListMaker(Account)	void
m	setFoodToBePurchased(LinkedList<Food>)	void



ShoppingCart		
f	shoppingCart	LinkedList<PurchaseList>
m	ShoppingCart()	
m	getShoppingCart()	LinkedList<PurchaseList>
m	setShoppingCart(LinkedList<PurchaseList>)	void
m	addPurchaseListToCart(PurchaseList)	void
m	getPurchaseListFromCart(PurchaseList)	PurchaseList
m	deletePurchaseListFromCart(PurchaseList)	void
m	updatePurchaseListInCart(PurchaseList, Account, String, LinkedList<Food>)	void

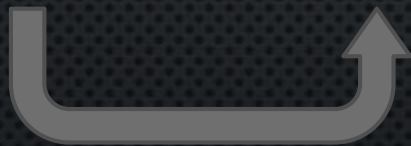


BENACHRICHTIGUNGEN KONFIGURIEREN

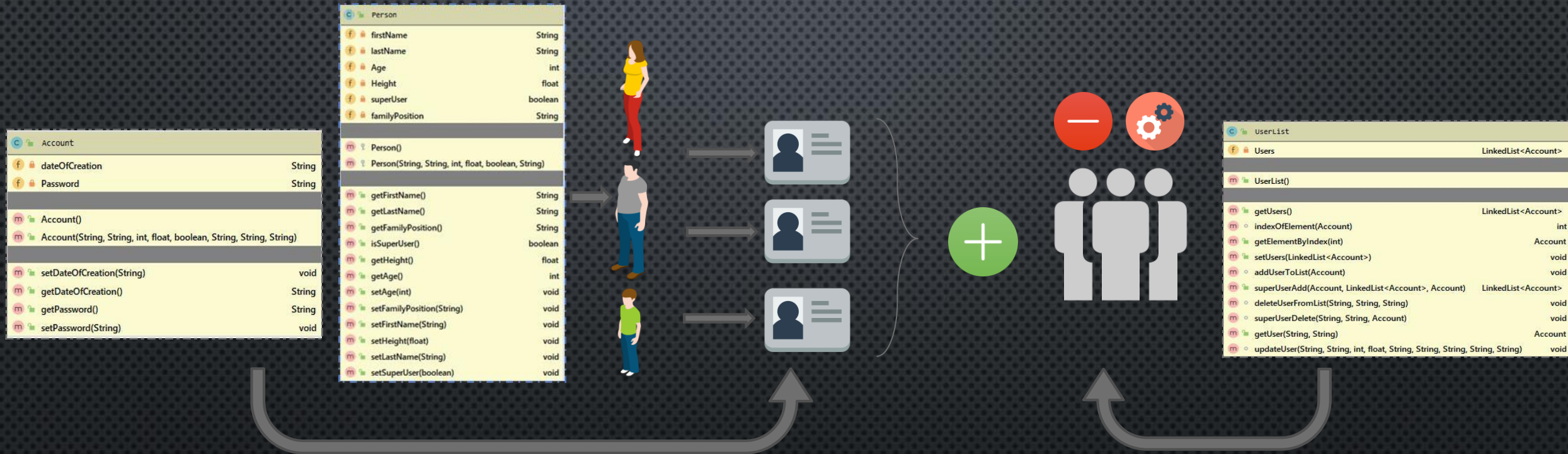
	NotificationFood	
	minimumQuantity	int
	selectedFood	Food
	dateOfExpiration	String
	NotificationFood()	
	NotificationFood(int, Food, String)	
	getSelectedFood()	Food
	getMinimumQuantity()	int
	setMinimumQuantity(int)	void
	setSelectedFood(Food)	void
	getDateOfExpiration()	String
	setDateOfExpiration(String)	void



	NotificationFoodList	
	Notifications	LinkedList<NotificationFood>
	NotificationFoodList()	
	getNotifications()	LinkedList<NotificationFood>
	setNotifications(LinkedList<NotificationFood>)	void
	addNotificationToList(NotificationFood)	void
	deleteNotificationFromList(NotificationFood)	void
	updateNotificationInList(NotificationFood, String, int, Food)	void
	getNotificationFromList(NotificationFood)	NotificationFood



BENUTZER VERWALTEN



VIELEN DANK FÜR DIE AUFMERKSAMKEIT